

# Introduction to MaBoSS and Continuous time Boolean modelling

Vincent Noël<sup>1</sup>

<sup>1</sup>Computational Systems Biology of Cancer,  
Institut Curie, INSERM U900, Mines ParisTech, PSL Research University, Paris

Computational Systems Biology for Complex Human Disease  
*from static to dynamic representations of disease mechanisms*

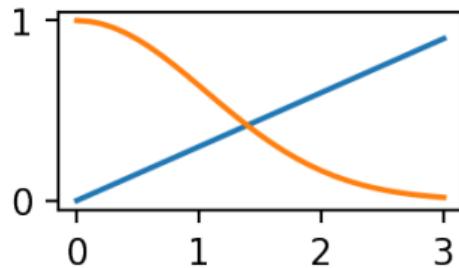
December 7th, 2022

- › MaBoSS : Continuous time boolean modelling
- › Example on Cohen's model of tumor invasion
- › Example on Montagud's model of prostate
- › Using MaBoSS : WebMaBoSS
- › Using MaBoSS : pyMaBoSS
- › MaBoSS Ecosystem

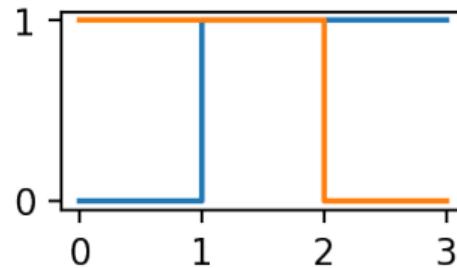
# MaBoSS : Continuous time boolean modelling

# MaBoSS : Continuous time boolean modelling

**Quantitative**



**Qualitative**



- › Values can be any quantity
- › Continuous time
- › Difficult to write
- › Difficult to simulate large models
- › Values are true/false
- › Sequences of events
- › Easy to write
- › Can simulate large models

# MaBoSS : Continuous time boolean modelling

Methodology article | [Open Access](#) | Published: 29 August 2012

## Continuous time boolean modeling for biological signaling: application of Gillespie algorithm

[Gautier Stoll](#)  , [Eric Viara](#), [Emmanuel Barillot](#) & [Laurence Calzone](#)

[BMC Systems Biology](#) 6, Article number: 116 (2012) | [Download Citation](#) 

6306 Accesses | 32 Citations | 1 Altmetric | [Metrics](#) 



Gautier Stoll



Eric Viara



Laurence Calzone



Emmanuel Barillot

# MaBoSS : Continuous time boolean modelling

**Markovian Boolean Stochastic Simulator**



<https://maboss.curie.fr/>

- › Boolean
- › Stochastic
- › Physical time
- › Handle different time scale processes (transcription, phosphorylation, etc.)

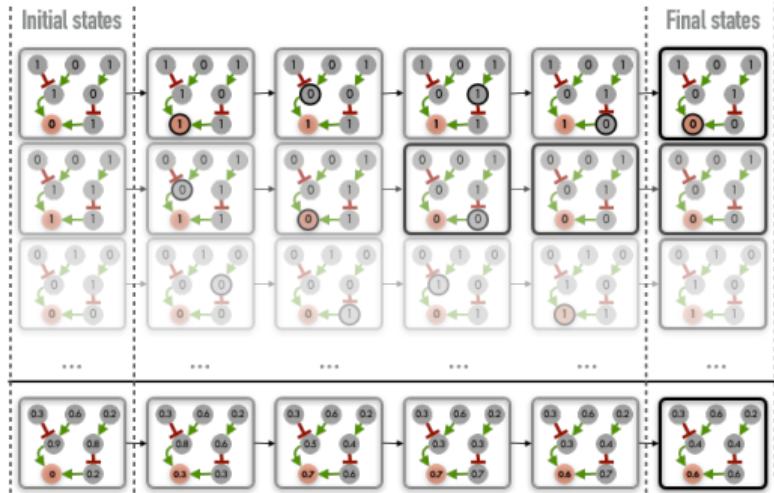
⇒ Fills the gap between ODE and Boolean modeling

# MaBoSS : Continuous time boolean modelling

- › Simulate transitions from a boolean network state to another using a markov chain
- › Transition rate :

$$p(S \rightarrow S') = \begin{cases} R_{up}(S), & \text{if } S_i = 0 \\ R_{down}(S), & \text{if } S_i = 1 \end{cases}$$

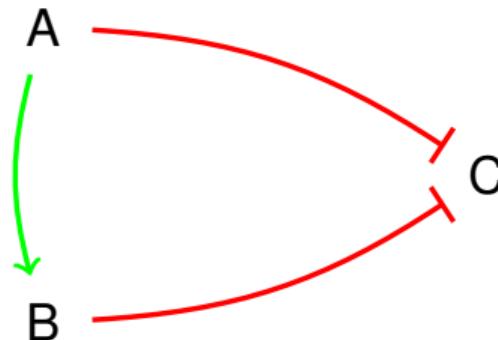
- › These transitions are stochastic : multiple trajectories are possible



From a set of simulated trajectories, we compute mean trajectories:  
⇒ We obtain probabilities per state over time

# MaBoSS : Continuous time boolean modelling

- › Toy example



## Continuous time Boolean modelling

### Formulas:

- ›  $A.R_{up} = 1$
- ›  $B.R_{up} = A ? 1 : 0$
- ›  $C.R_{up} = (\neg A \mid \neg B) ? 1 : 0$
- ›  $A.R_{down} = 0$
- ›  $B.R_{down} = A ? 0 : 1$
- ›  $C.R_{down} = (\neg A \mid \neg B) ? 0 : 1$

### Initial values:

- ›  $A_0 = 1$
- ›  $B_0 = 0$
- ›  $C_0 = 1$

# MaBoSS : Continuous time boolean modelling

## › Model definition : BND file

- › Definition of activation and inactivation rates
- › Allows complex cases : multiple rates for multiple states, ...
- › Possibility to use parameters defined in the simulation settings
- › Now allows other boolean model formats : SBML-qual, BNet

```
1 node A {  
2     rate_up = 1;  
3     rate_down = 0;  
4 }  
5  
6 node B {  
7     logic = A;  
8     rate_up = @logic ? $act_B : 0;  
9     rate_down = 0;  
10 }  
11  
12 node C {  
13     logic = !A | !B;  
14     rate_up = @logic ? $act_C : 0;  
15     rate_down = @logic ? 0 : $act_C;  
16 }
```

# MaBoSS : Continuous time boolean modelling

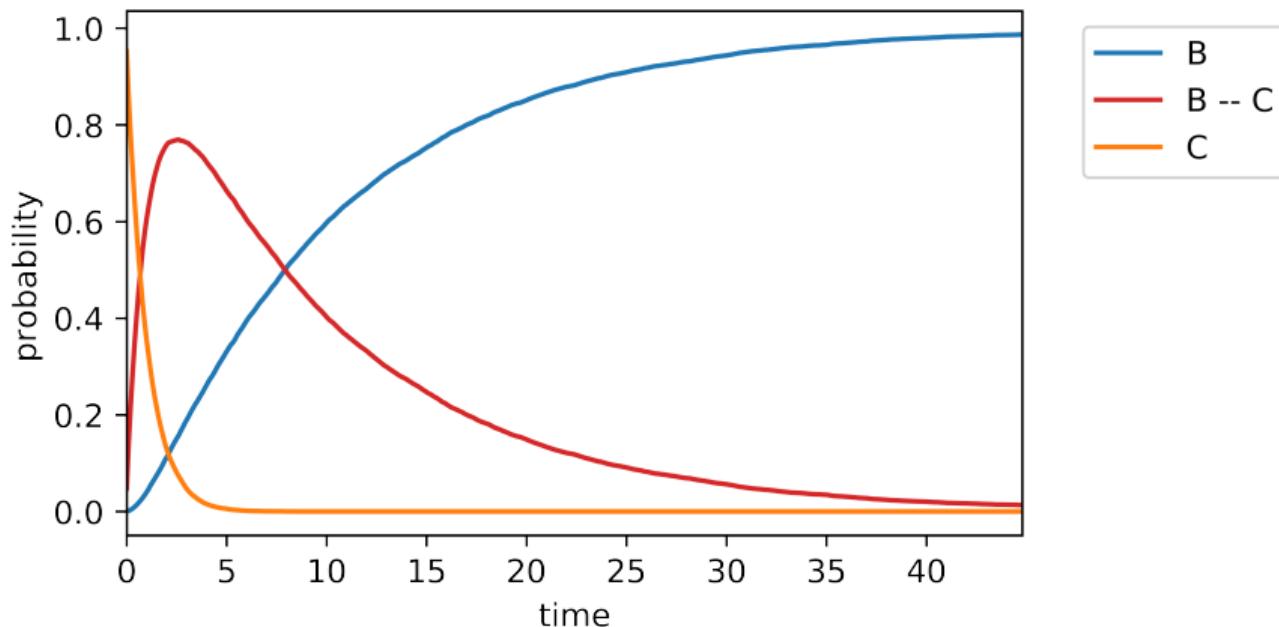
## › Simulation settings : CFG file

- › Initial state of nodes (Fixed, Stochastic)
- › Internal/Output variables
- › Parameters
- › Settings

```
1 A.istate = TRUE;
2 B.istate = FALSE;
3 C.istate = TRUE;
4
5 A.is_internal = TRUE;
6
7 $act_B = 1;
8 $act_C = 0.1;
9
10 max_time = 45;
11 sample_count = 10000;
12 use_physrandgen = 1;
13 thread_count = 1;
```

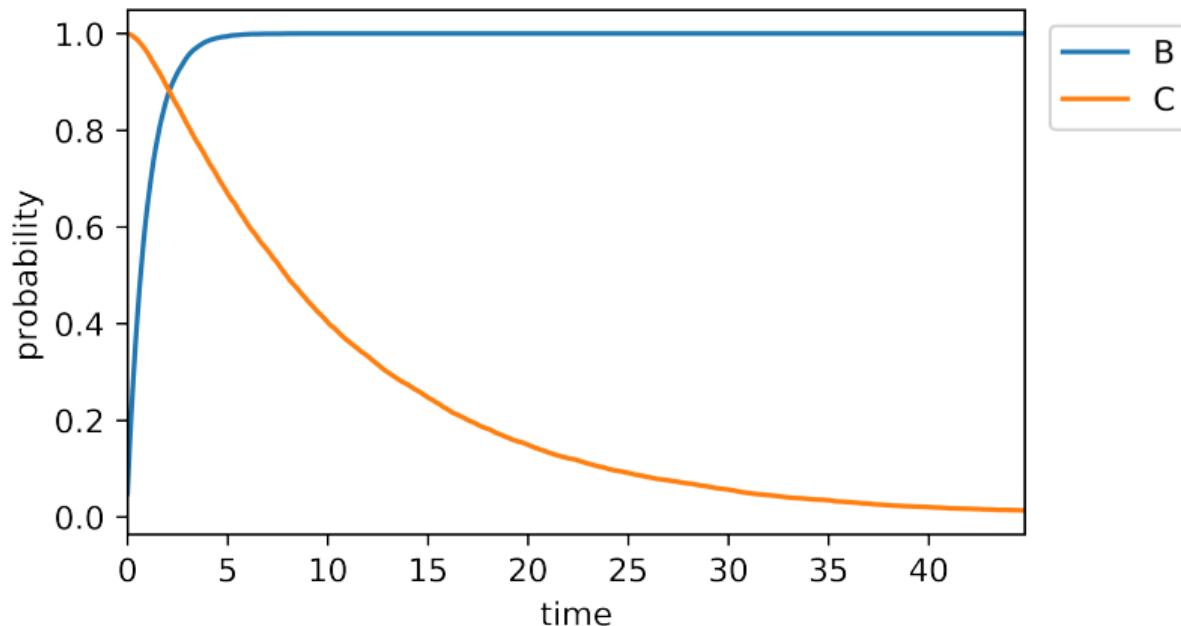
# MaBoSS : Continuous time boolean modelling

- State probability trajectories



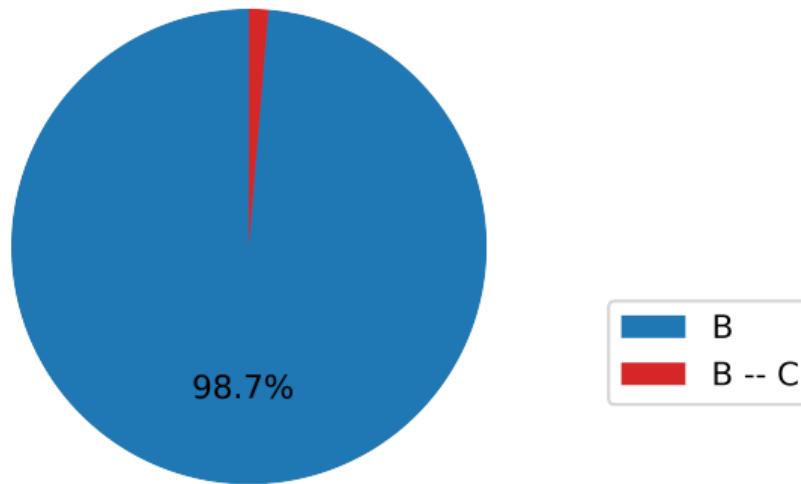
# MaBoSS : Continuous time boolean modelling

- › Node probability trajectories



# MaBoSS : Continuous time boolean modelling

- › Final ( $\neq$  steady) state distribution



# Example on Cohen's model of tumor invasion

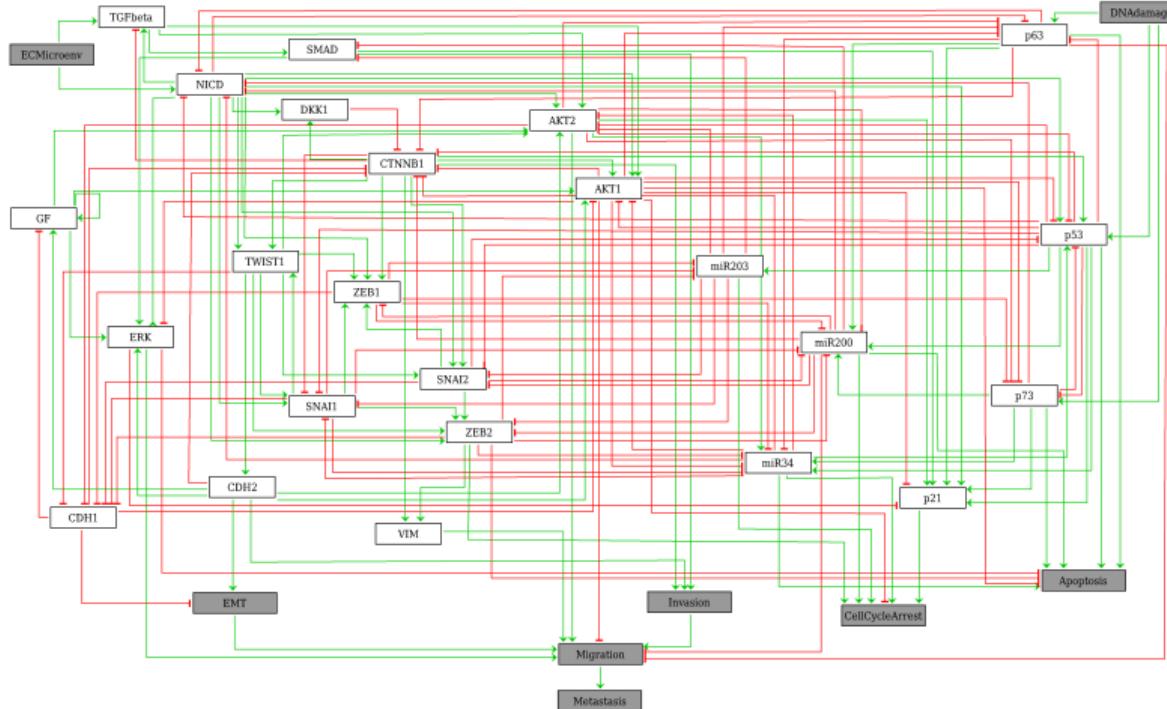
# Example on Cohen's model of tumor invasion

## Two input nodes:

- › ECMicroenv
- › DNADamage

## Six output nodes:

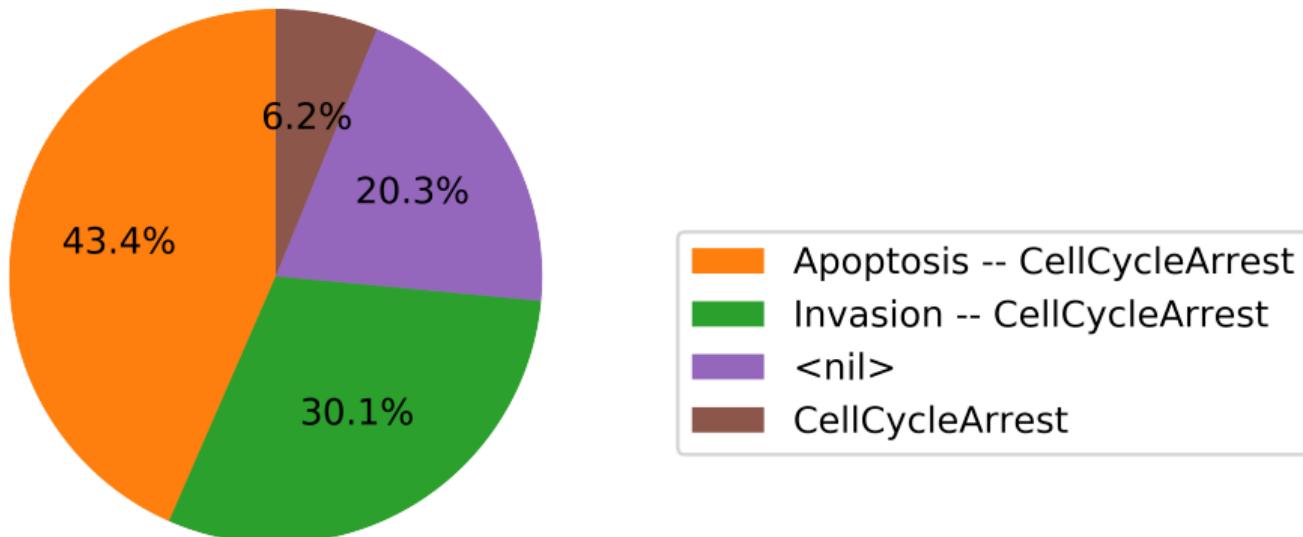
- › EMT
- › CellCycleArrest
- › Apoptosis
- › Invasion
- › Migration
- › Metastasis



Cohen et al. (2015) PLoS Computational Biology

## Example on Cohen's model of tumor invasion

- › Final state distribution

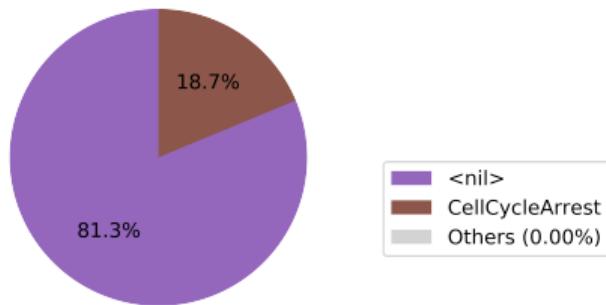


## Example on Cohen's model of tumor invasion

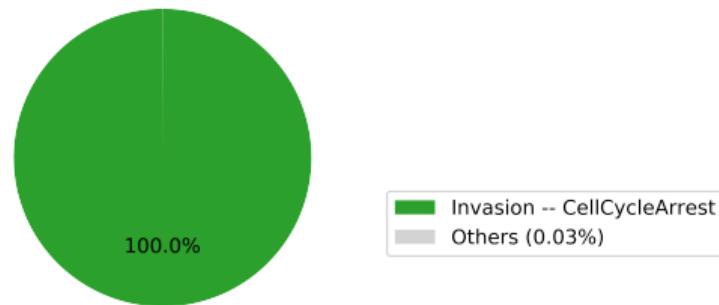
### › Influence of Microenvironment

Here we simulate with different initial states : with or without the microenvironment (and without DNA damages)

**No Microenvironment**



**Microenvironment**



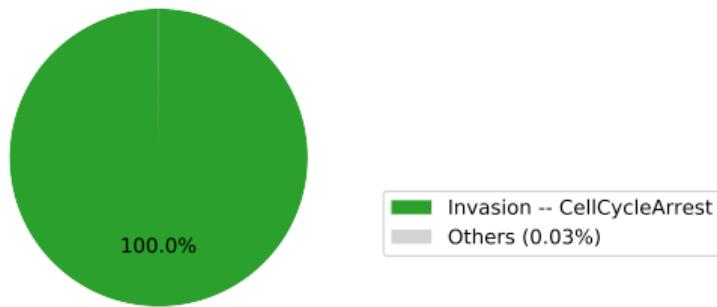
⇒ When we activate the microenvironment, we obtain an invasive phenotype

## Example on Cohen's model of tumor invasion

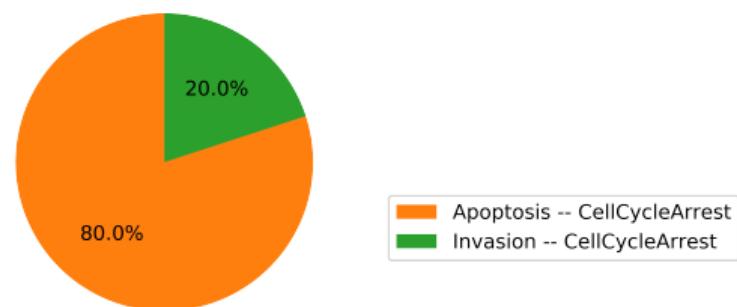
- Influence of DNA damage

Here again we modify the initial values : with and without DNA damages (with microenvironment)

**Microenvironment**



**Microenvironment + DNA damage**

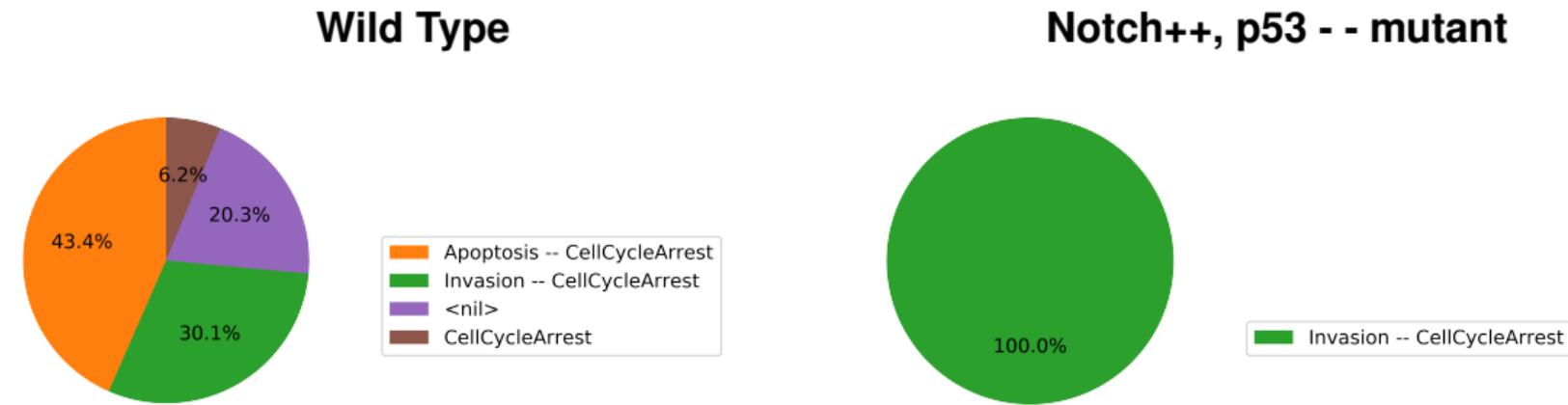


⇒ When we activate DNA damages, we are able to trigger apoptosis in 80% of the cells

## Example on Cohen's model of tumor invasion

- › Notch++, p53- - mutant

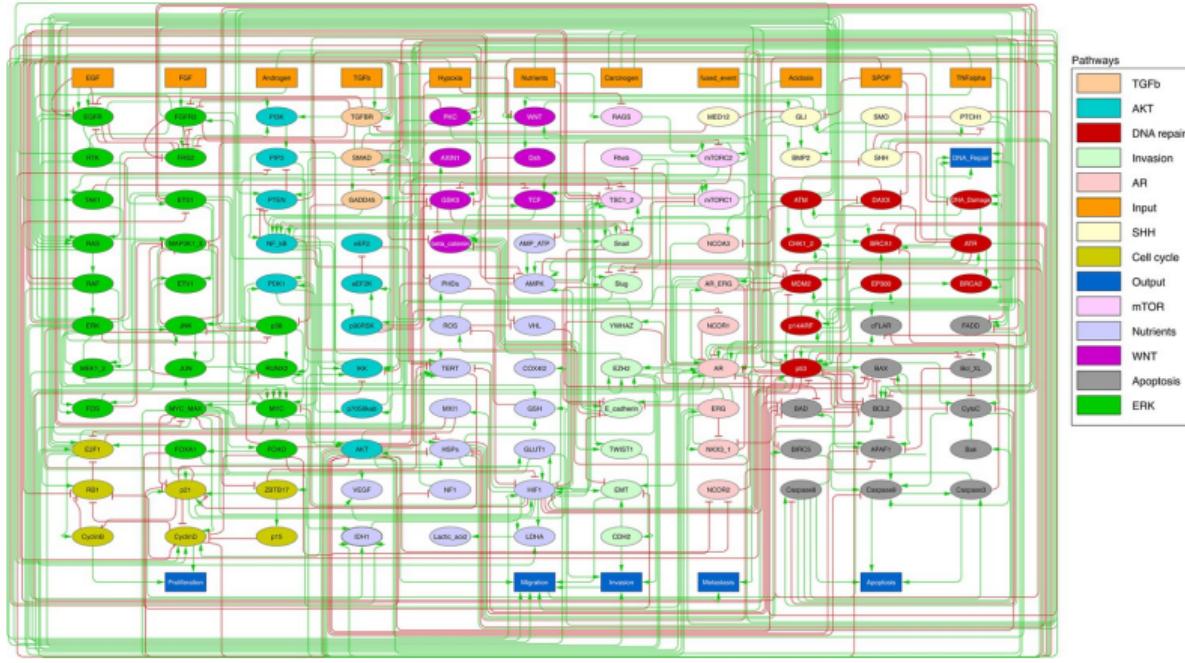
Here we modify the model to force the activity of Notch, and block the activity of p53



⇒ The results shows that whatever the inputs, we obtain a invasive phenotype

# Example on Montagud's model of prostate

# Example on Montagud's model of prostate

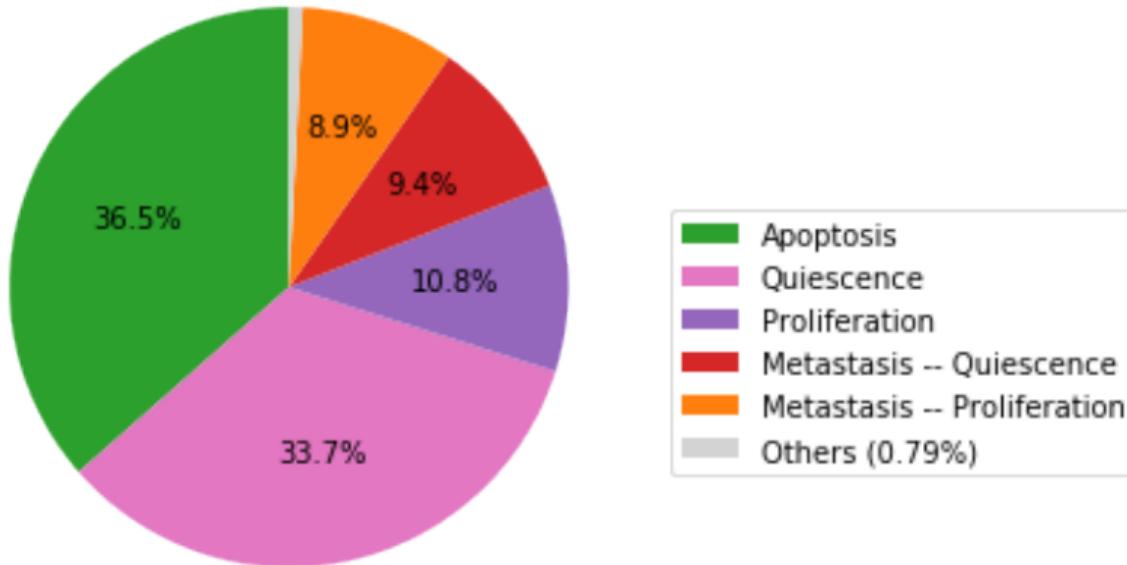


Montagud et al. (2022) eLife

*The model can be considered as a model of healthy prostate cells when no mutants (or fused genes) are present. We refer to this model as the wild type model.*

## Example on Montagud's model of prostate

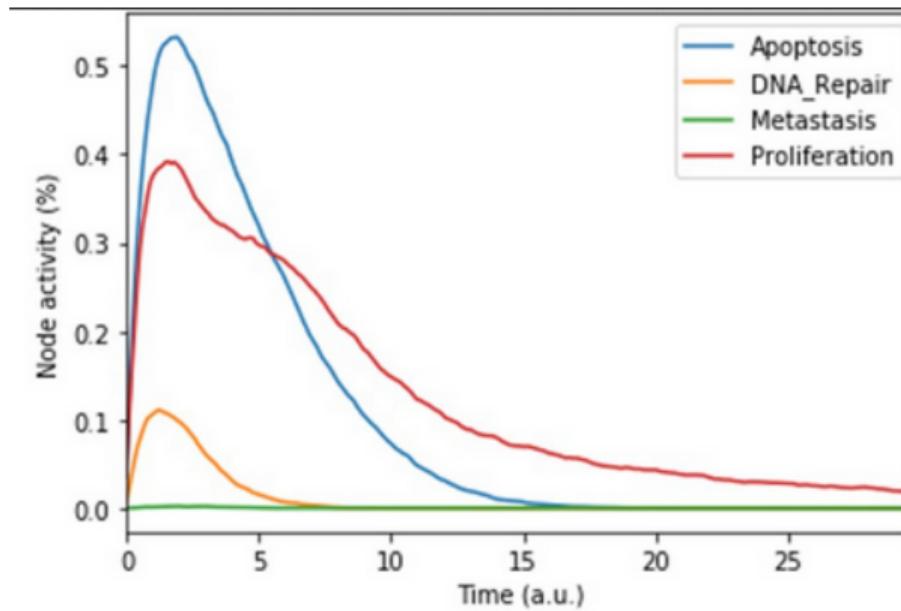
- Any possible input value



*This piechart show all possible behaviours of the model, and is based on simulation where all input combinations are tested.*

## Example on Montagud's model of prostate

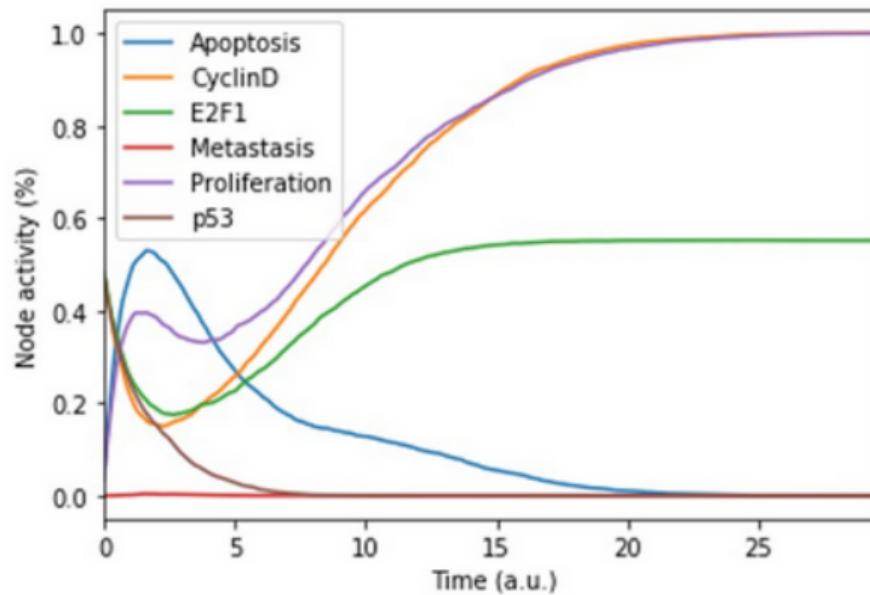
### › Wild Type Phenotype



*These healthy cells mostly exhibit quiescence (neither proliferation nor apoptosis) in the absence of any input.*

## Example on Montagud's model of prostate

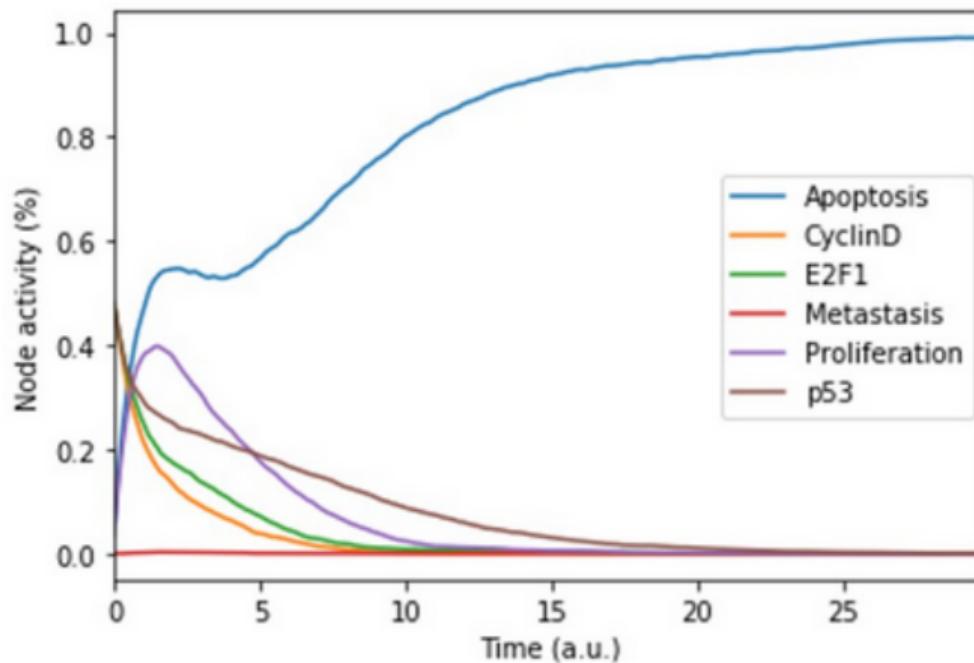
### › Proliferative Phenotype



*When Nutrients and growth factors (EGF or FGF) are present, Proliferation is activated.*

## Example on Montagud's model of prostate

### › Apoptotic Phenotype



*Cell death factors (such as Caspase 8 or 9) trigger Apoptosis in the absence of SPOP, while Hypoxia and Carcinogen facilitate apoptosis but are not necessary if cell death factors are present.*

# Using MaBoSS : WebMaBoSS



## WebMaBoSS

A web tool for simulating Boolean models

Click [here](#) to login if you already have an account, otherwise you can quickly create one [here](#).

If you want to quickly test WebMaBoSS, you can have a look at our [demo project](#).

### Model analysis

WebMaBoSS allows simulations, and multiple outputs for results. It also allows sensitivity analysis by performing single and double mutations.

### Compatibility

WebMaBoSS is able to import models in MaBoSS format (bnd, cfg files), BoolNet format, SBML-qual format, or in GINsim format. It also allows to export models in any of these three formats.

### Public databases

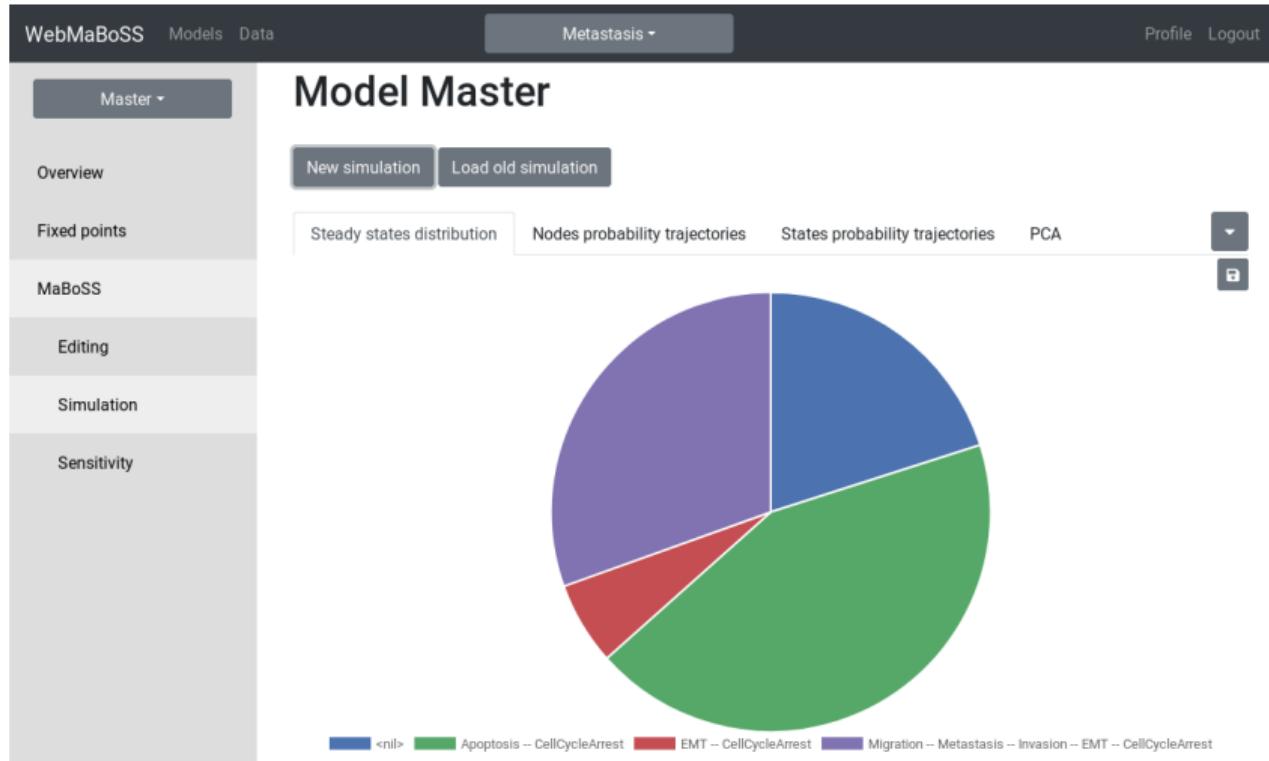
WebMaBoSS allows to browse models from CellCollective and BioModels, and import them.

WebMaBoSS was created and is maintained by the team [Computational System Biology of Cancer](#) at [Institut Curie](#).

It is open-source and available on [GitHub](#), where you can also find instructions to run it locally and tutorials.

<https://maboss.curie.fr/webmaboss/>

# Using MaBoSS : WebMaBoSS



# Using MaBoSS : WebMaBoSS

WebMaBoSS   Models   Tutorials   About   Contact   Help   Cancer models

Profile   Logout

## Import model

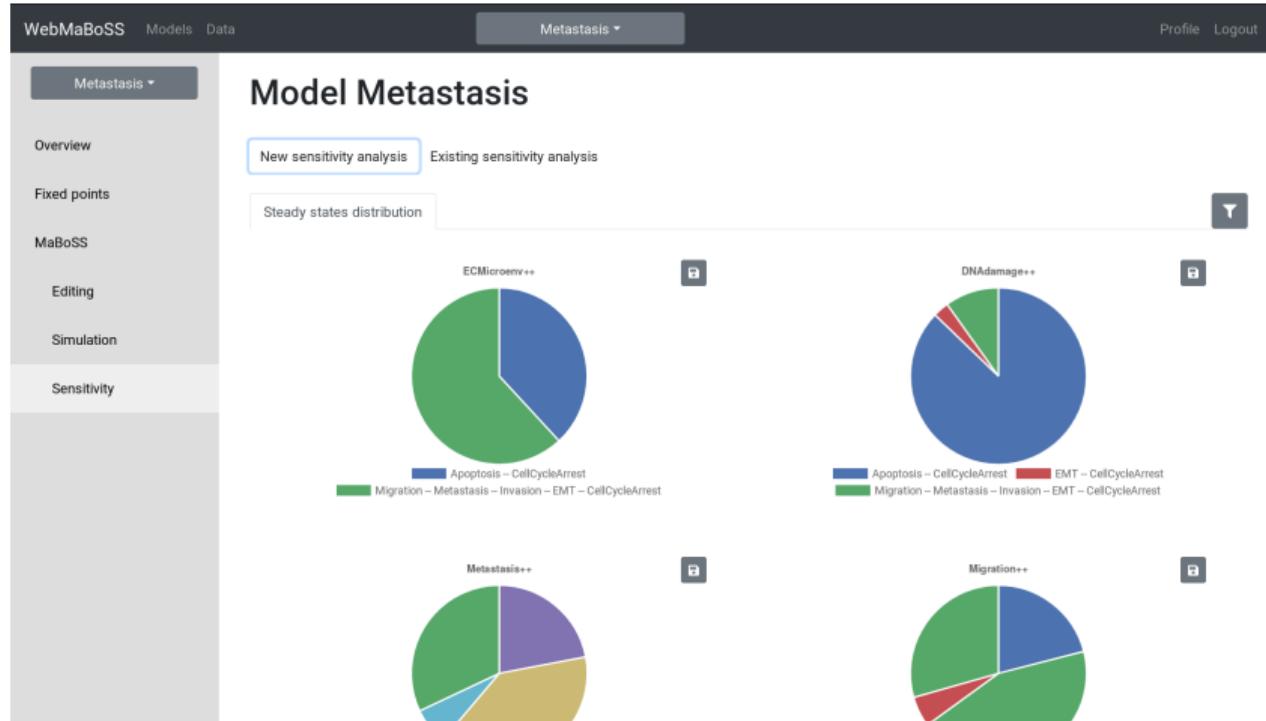
BioModels   CellCollective

Name	Author
Traynard2016 - Mammalian cell cycle regulation - Logical Model	Pauline Traynard
Cacace2020 - Logical model of the regulatory network controlling...	Denis Thieffry
Howell2020- Compartmental Logical model of mitotic exit	Rowan Howell
Terfve2012 - Signalling in liver cancer - logical model	Vijayalakshmi Chelliah
Floc'hlay2020 - SeaUrchin_model_ginsim	Krishna Kumar Tiwari
Mbodj2016 - Mesoderm specification during Drosophila develop...	Denis Thieffry
Rodriguez-Jorge2019 - Boolean model of TCR signaling for CD4...	Denis Thieffry
Rodriguez-Jorge2019 - Boolean model of combined TCR and TL...	Denis Thieffry
Afenya2018 - peripheral blood dynamics in the disease state	Szeyi Ng

Use SBML names

**Close**

# Using MaBoSS : WebMaBoSS



# Using MaBoSS : pyMaBoSS

- › Initially developped by Nicolas Levy
- › Maintained by Aurelien Naldi, Loic Pauleve, me  
<https://github.com/colomoto/pyMaBoSS>
- › Available on Pypi:  
\$ pip install maboss
- › Available on Conda:  
\$ conda install -c colomoto pymaboss



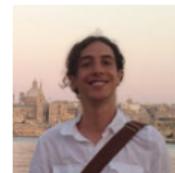
Nicolas Levy



Aurelien Naldi



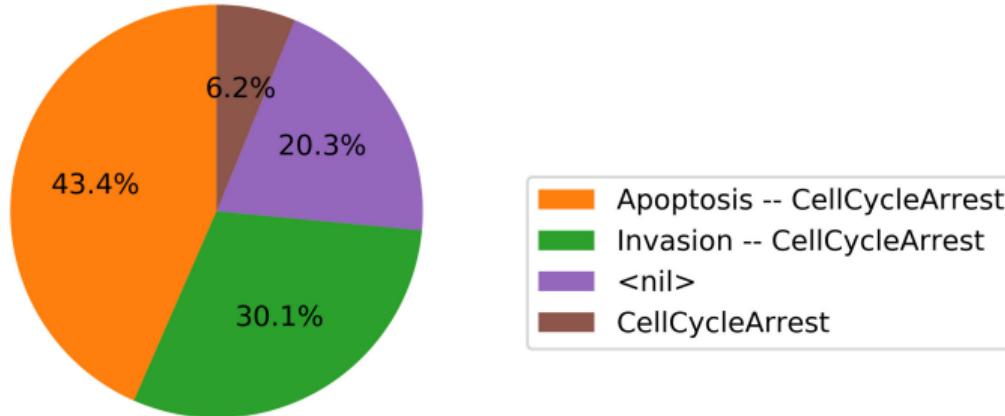
Lo c Paulev 



Vincent No l

## Using MaBoSS : pyMaBoSS

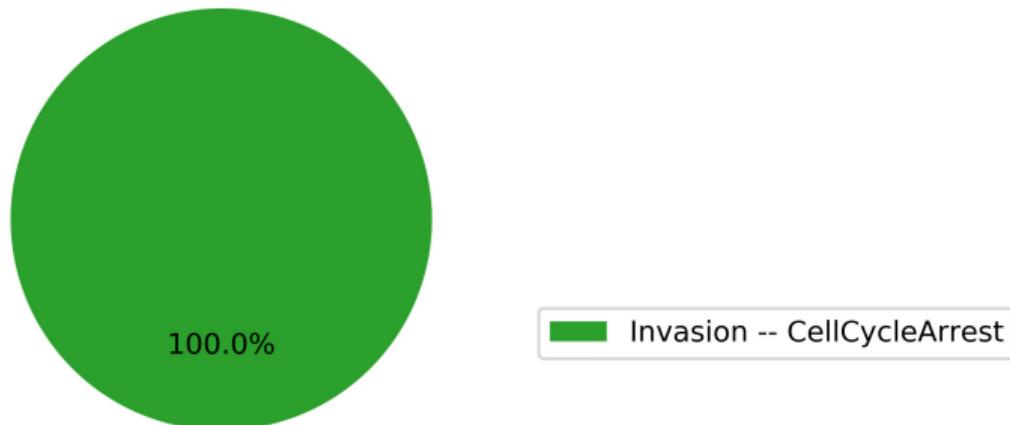
```
import maboss
model = maboss.load("metastasis.bnd", "metastasis.cfg")
res = model.run()
res.plot_piechart()
```



- › A few lines of code to simulate the model and plot the steady states distribution

## Using MaBoSS : pyMaBoSS

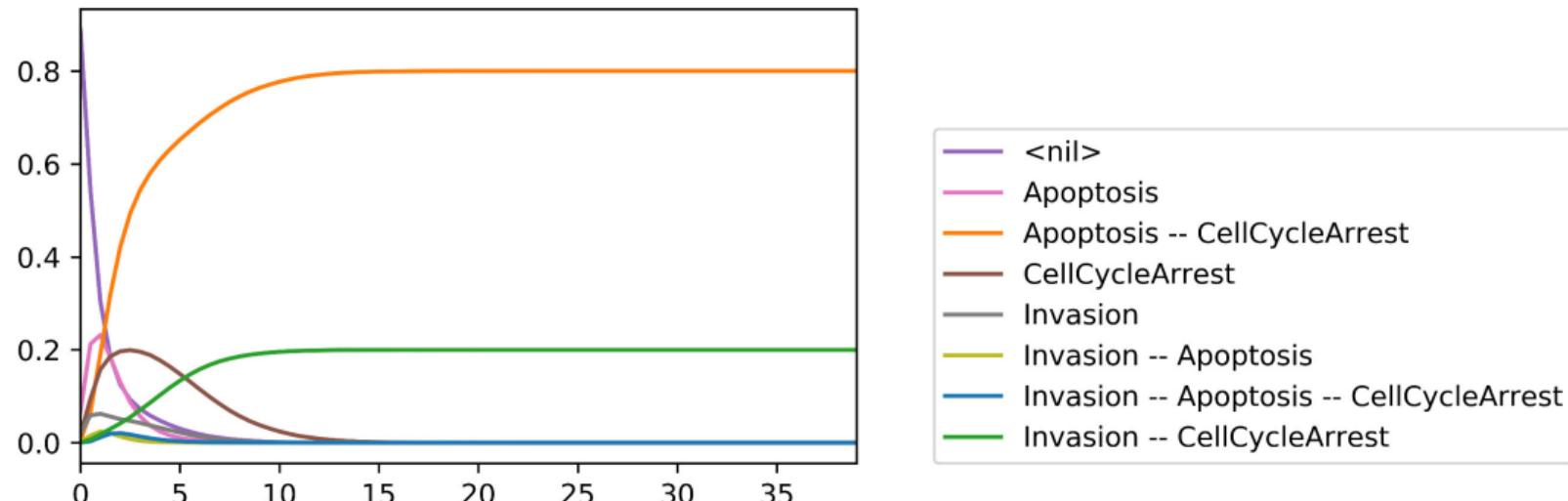
```
model_mutant = model.copy()
model_mutant.mutate('NICD','ON')
model_mutant.mutate('p53','OFF')
res_mutant = model_mutant.run()
res_mutant.plot_piechart()
```



- › Performing mutant simulations

# Using MaBoSS : pyMaBoSS

```
res.plot_trajectory()
```



- › Visualizing state probability trajectories

## Using MaBoSS : pyMaBoSS

```
res.get_nodes_probtraj()
```

	Apoptosis	CellCycleArrest	EMT	Invasion	Metastasis	Migration
0.0	0.009598	0.000902	0.000000	0.005312	0.000000	0.000000
0.2	0.050760	0.011256	0.000233	0.028856	0.000000	0.000000
0.4	0.106823	0.037696	0.001628	0.063692	0.000000	0.000000
0.6	0.165967	0.085195	0.004456	0.099778	0.000000	0.000000
0.8	0.221066	0.138753	0.010313	0.125779	0.000000	0.000000
1.0	0.265365	0.195669	0.018824	0.145638	0.000000	0.000098
1.2	0.300923	0.255950	0.030564	0.158304	0.000000	0.000100
1.4	0.330531	0.313228	0.045127	0.169356	0.000006	0.000188

- › Getting the results as Panda dataframes

# Using MaBoSS : pyMaBoSS



PROTOCOLS  
published: 06 July 2018  
doi: 10.3389/fphys.2018.00787



## Prediction of Mutations to Control Pathways Enabling Tumor Cell Invasion with the CoLoMoTo Interactive Notebook (Tutorial)

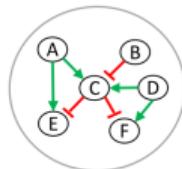
Nicolas Levy<sup>1,2</sup>, Aurélien Naldi<sup>3</sup>, Céline Hernandez<sup>3</sup>, Gautier Stoll<sup>4,5,6,7,8</sup>, Denis Thieffry<sup>3</sup>, Andrei Zinovyev<sup>9,10,11,12</sup>, Laurence Calzone<sup>9,10,11</sup> and Loïc Paulevé<sup>1\*</sup>

- › Python Notebook with Logical Modelling tools
- › Reproducible research

# MaBoSS Ecosystem

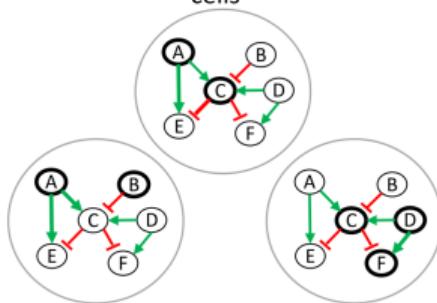
# MaBoSS Ecosystem

**MaBoSS**  
Model of an individual cell



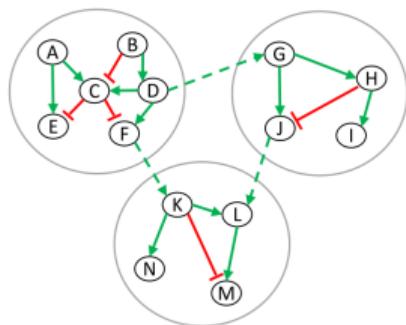
**EnsembleMaBoSS**

Ensembles of models of non interacting cells



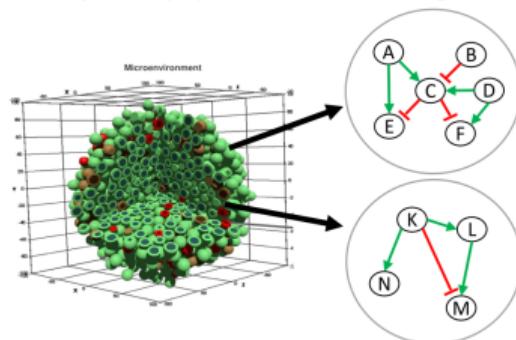
**UPMaBoSS**

Model of dynamic populations of interacting cells



**PhysiBoSS**

Model of spatial organization of dynamic population of interacting cells



# MaBoSS Ecosystem

- › EnsembleMaBoSS

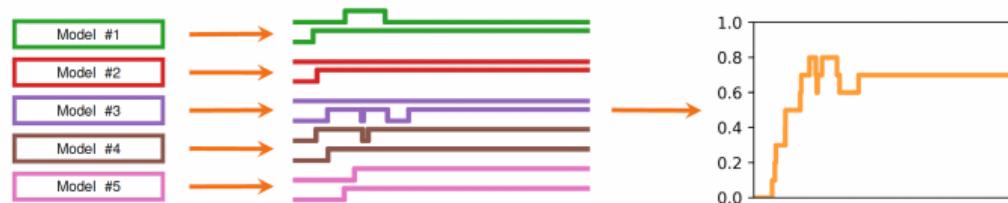
**Problem:** Sometimes we can't decide which boolean formula is the right one

**The principle:** Define constraints and use Answer-Set-Programming to produce models that satisfy them

**BoNesis** (Chevalier et al.)

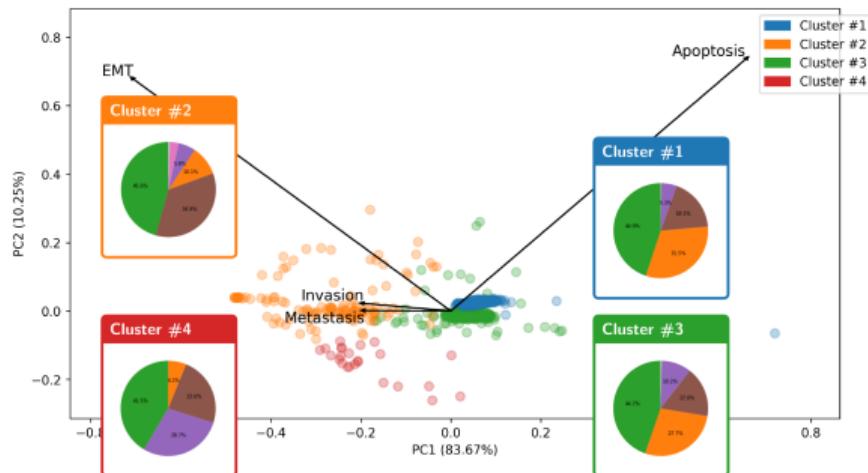
- › Start from an interaction graph (network without rules)
- › Logical rules are based in incoming interactions
- › Models must satisfy some biological properties (stable states)
- › Stable states must be reachable from set of initial conditions

**EnsembleMaBoSS** Models created by BoNesis can be run with MaBoSS

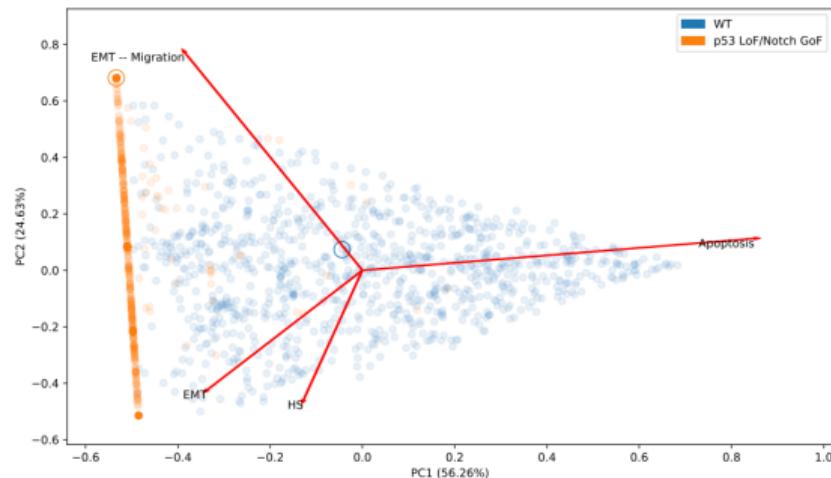


# MaBoSS Ecosystem

## › EnsembleMaBoSS



⇒ Select a subset of models based on their steady state distribution

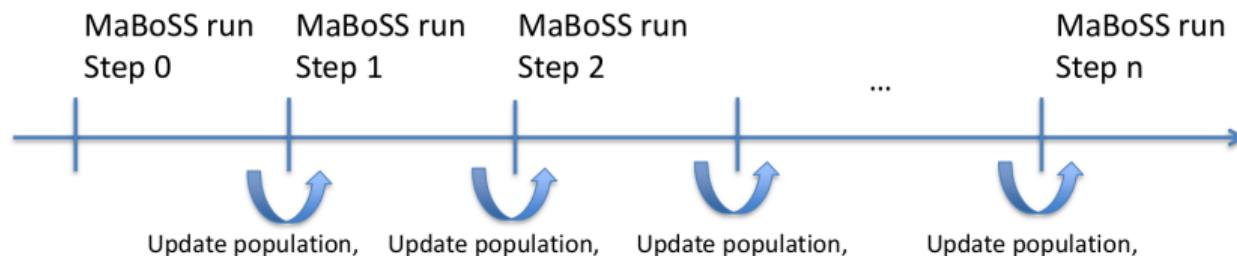


⇒ Analyse the composition of the ensemble before/after a treatment

# MaBoSS Ecosystem

- › UPMaBoSS : Population models

- › We start from an existing MaBoSS model
- › We add model nodes for division and death
- › We add inter-cellular communication
- › Chaining MaBoSS simulations, updating population and communication between each



# MaBoSS Ecosystem

## › UPMaBoSS : Population models

Cell division



$p(\text{divide}) \times 2$



$$\text{population} = \text{population} * (1 + p(\text{divide}))$$

Cell death



$p(\text{dead}) = 0$



$$\text{population} = \text{population} * (1 - p(\text{dead}))$$

Intercellular communication

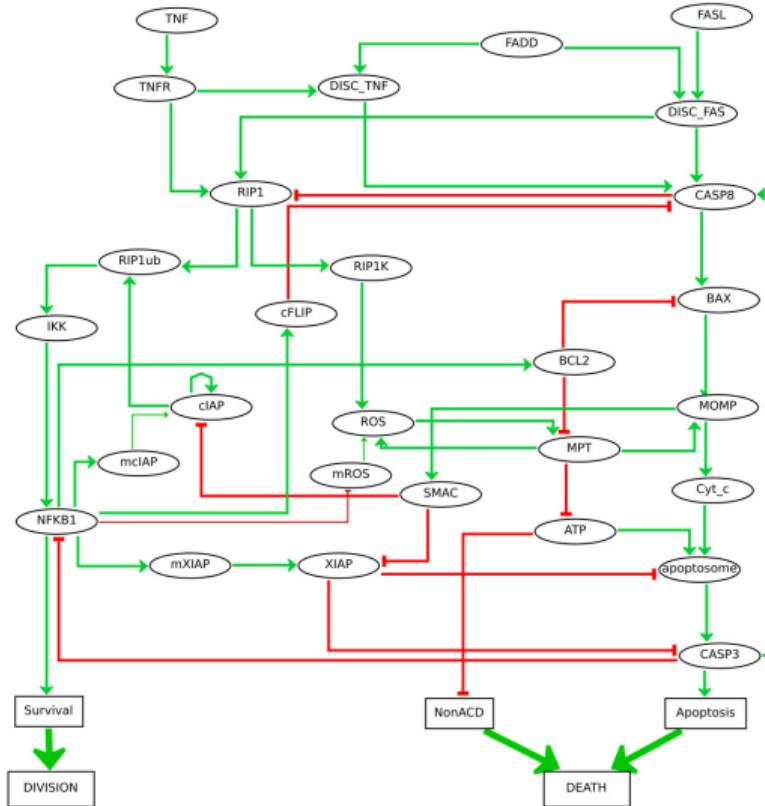
Update the environmental condition of other cells

# MaBoSS Ecosystem

- UPMaBoSS : Population models

We map the cell fates to population updates :

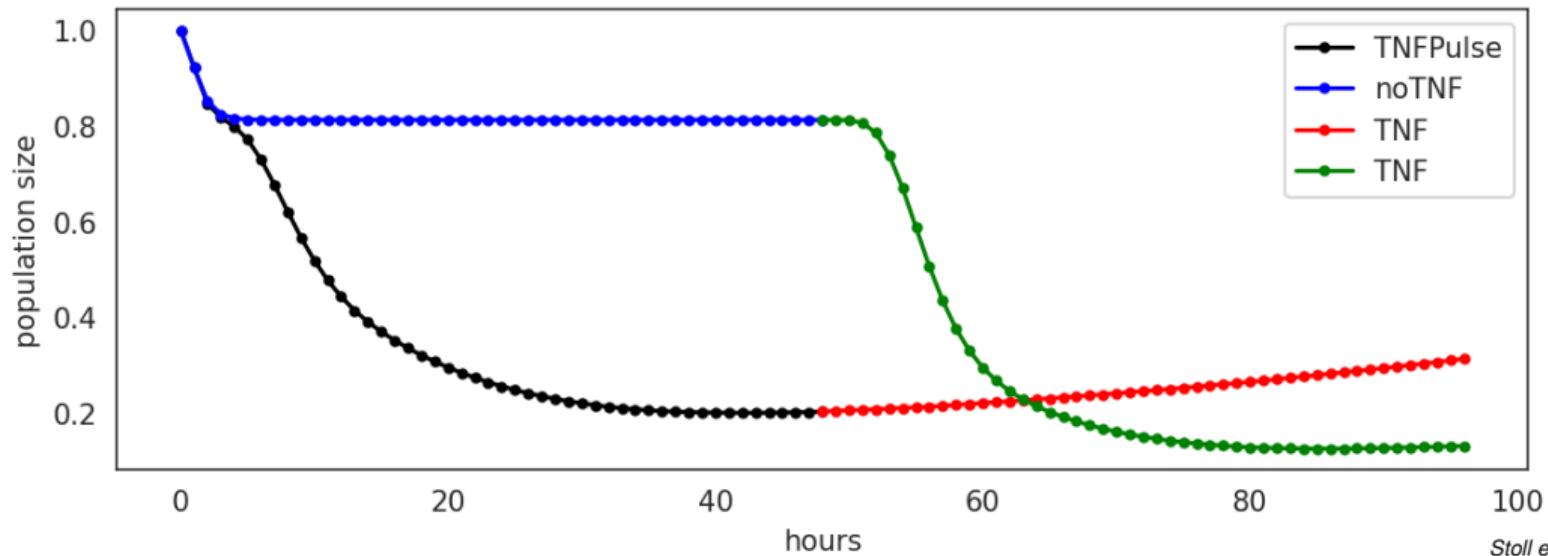
- Survival node : Cell division
- Apoptosis and NonACD : Cell death



Adapted from Stoll et al., 2022

# MaBoSS Ecosystem

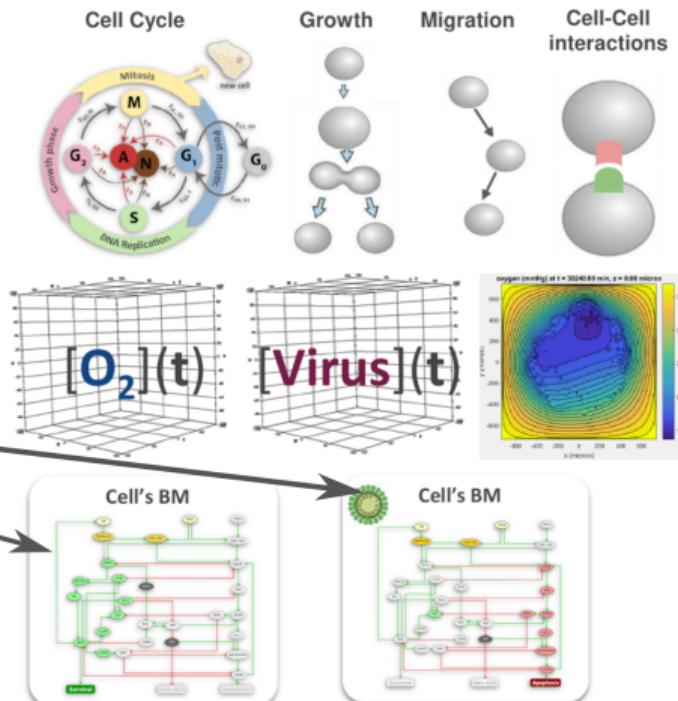
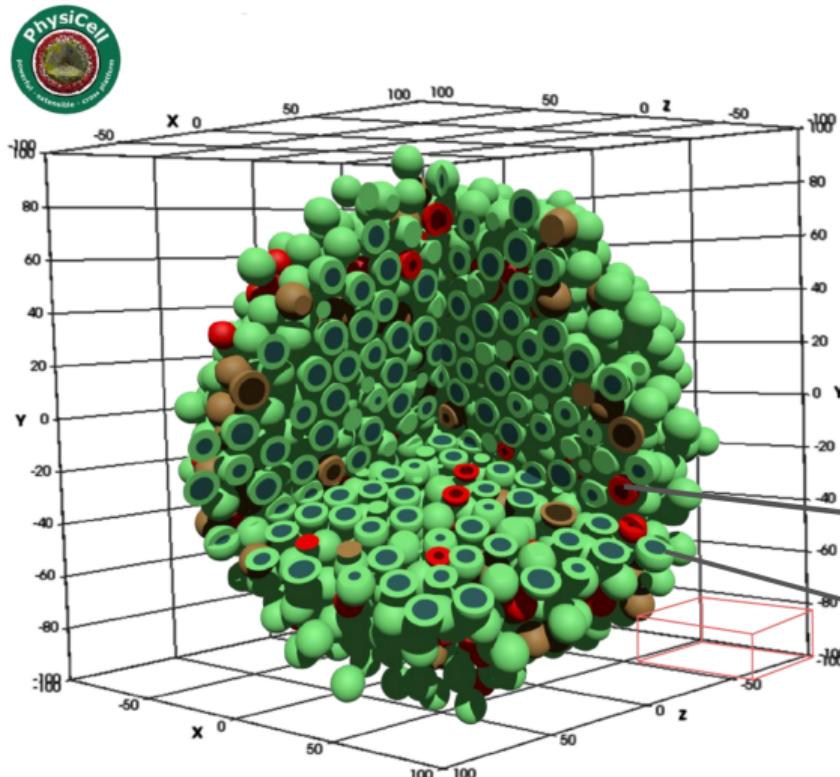
- › UPMaBoSS : Population models



- › Study the resistance to TNF treatment

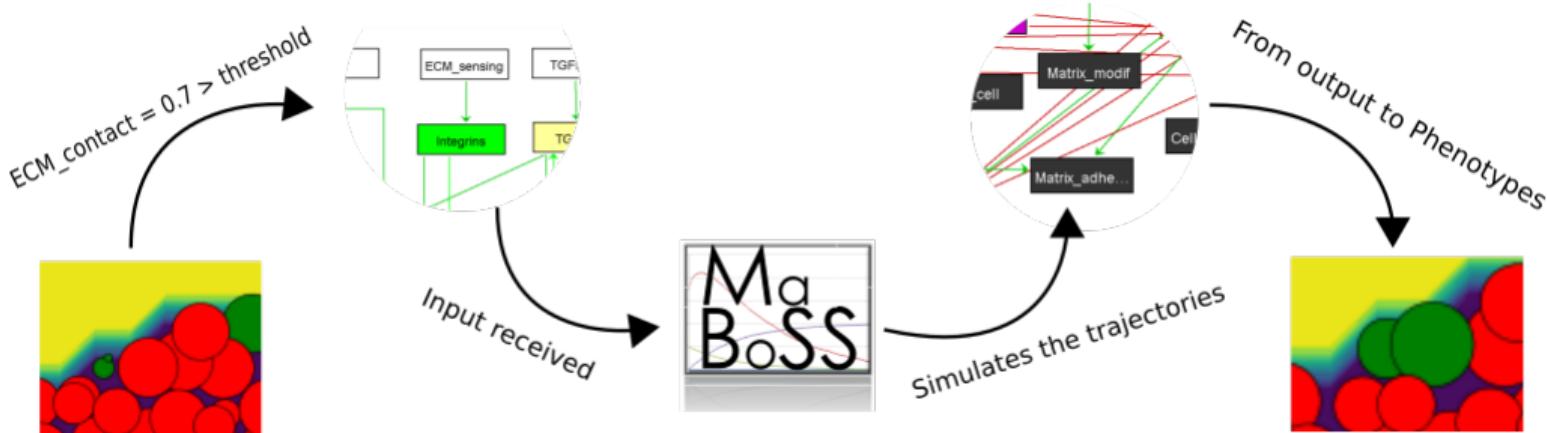
# MaBoSS Ecosystem

## ► PhysiBoSS : Agent-based models



# MaBoSS Ecosystem

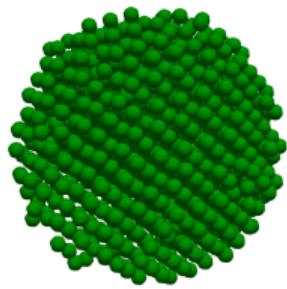
- › PhysiBoSS : Agent-based models



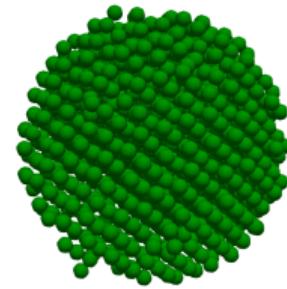
- › We take signals as inputs, and modify behaviors according to outputs
- › Each cell runs one single (stochastic) simulation every *intracellular\_dt*

# MaBoSS Ecosystem

- › PhysiBoSS : Agent-based models



Sustained TNF treatment



Pulses of TNF treatment

# Acknowledgments



## MaBoSS early team



Gautier Stoll



Eric Viara



Laurence Calzone



Emmanuel Barillot

## MaBoSS ecosystem team



Aurelien Naldi



Denis Thieffry



Loïc Paulevé



Nicolas Levy



Stéphanie  
Chevalier



Arnaud  
Montagud



Vincent Noël



Gaëlle Letort



Mihaly Koltai



Andrei  
Zinovyev

<https://maboss.curie.fr>  
[vincent.noel@curie.fr](mailto:vincent.noel@curie.fr)